THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Spectral bounds for graph partitioning with prescribed partition sizes

Citation for published version:
Anjos, M & Neto, J 2019, 'Spectral bounds for graph partitioning with prescribed partition sizes', *Discrete Applied Mathematics*, vol. 269, pp. 200-210. https://doi.org/10.1016/j.dam.2019.06.007

Digital Object Identifier (DOI):
10.1016/j.dam.2019.06.007

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Discrete Applied Mathematics

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

OPEN ACCESS

Download date: 07. May. 2023

Spectral bounds for graph partitioning with prescribed partition sizes ☆

Miguel F. Anjos[a,b], José Neto[c,*]

[a] *School of Mathematics, University of Edinburgh, James Clerk Maxwell Building, Peter Guthrie Tait Road, Edinburgh EH9 3FD, United Kingdom*
[b] *GERAD & Polytechnique Montréal, QC, Canada H3C 3A7*
[c] *Samovar, CNRS, Télécom SudParis, Université Paris-Saclay, 9 rue Charles Fourier, 91011 Evry, France*

Abstract

Given an undirected edge weighted graph, the graph partitioning problem consists in determining a partition of the node set of the graph into subsets of prescribed sizes, so as to maximize the sum of the weights of the edges having both endpoints in the same subset. We introduce a new class of bounds for this problem relying on the full spectral information of the weighted adjacency matrix A. The expression of these bounds involves the eigenvalues and particular geometrical parameters defined using the eigenvectors of A. A connection is established between these parameters and the maximum cut problem. We report computational results showing that the new bounds compare favorably with previous bounds in the literature.

Keywords: Graph partitioning, Adjacency matrix eigenvalues, Adjacency matrix eigenvectors, Maximum cut, Semidefinite programming

☆This work was supported by a public grant as part of the Investissement d'avenir project, reference ANR-11-LABX-0056-LMH, in a joint call with Gaspard Monge Program for optimization, operations research and their interactions with data sciences.

*Corresponding author
Email addresses: `anjos@stanfordalumni.org` (Miguel F. Anjos), `Jose.Neto@telecom-sudparis.eu` (José Neto)

1. Introduction

Let $G = (V, E)$ be an undirected simple graph having node set $V = \{1, 2, \ldots, n\}$, edge set E, and let $w \in \mathbb{R}^E$ denote a weight function on the edges. Let k denote a positive integer, $k < n$, and $m = (m_1, m_2, \ldots, m_k)^T$ denote a vector of k positive integers satisfying $\sum_{i=1}^{k} m_i = n$ and $m_1 \geq m_2 \geq \ldots \geq m_k$. We consider the problem denoted by $\mathcal{P}_m^k$ which consists in determining a partition of V into k subsets $S_1, S_2, \ldots, S_k$ of sizes $m_1, m_2, \ldots, m_k$, respectively, so as to maximize the sum of the weights of the edges having both endpoints in the same subset of the partition. This NP-hard problem [17] has applications, e.g. in microchip design [8, 25], computer program segmentation [12], the design of power networks [19] and other layout problems [20]. Before presenting some related work, we introduce some useful notation.

Given a positive integer q, let $[q]$ denote the set of integers $\{1, 2, \ldots, q\}$. Let $W = (w_{ij})_{(i,j)\in[n]^2}$ stand for the weighted adjacency matrix of G: $W_{ij} = w_{ij}$ if $ij \in E$ and $W_{ij} = 0$ otherwise. Given two disjoint node subsets A, B, let $w[A, B]$ denote the sum of the weights of the edges having one endpoint in A and the other in B:

$$w[A, B] = \sum_{(i,j)\in A\times B:\ ij\in E} w_{ij}.$$

Similarly, $w[A]$ denotes the sum of the weights of the edges with both endpoints in A:

$$w[A] = \sum_{(i,j)\in A^2:\ ij\in E, i<j} w_{ij}.$$

Given any matrix $M \in \mathbb{R}^{n\times n}$, let $\lambda_1(M) \geq \lambda_2(M) \geq \ldots \geq \lambda_n(M)$ denote its eigenvalues in nonincreasing order. For the particular case when $M = W$, we shall more simply use λ_i instead of $\lambda_i(W)$, for all $i \in [n]$. Also, let $\nu_1, \nu_2, \ldots, \nu_n$ stand for unit and pairwise orthogonal eigenvectors corresponding to the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$, respectively. Let Z^* denote the optimal objective value of $\mathcal{P}_m^k$.

Approaches used to solve $\mathcal{P}_m^k$ and some of its variants include notably: heuristics [18], linear programming (polyhedral combinatorics) [9, 10] and semidefinite programming [1, 6, 13, 16, 21]. Alternatively, some research focused on establishing bounds on Z^* using eigenvalue-based techniques [14, 15, 23]. In this paper we pursue further investigations along the eigenvalue-based line of research. Before introducing our new results, we present related works from the literature.

The following upper bound on Z^* directly follows from Donath and Hoffman's work [14].

Proposition 1.1. *[14] Let $U \in \mathbb{R}^{n\times n}$ denote any diagonal matrix such that the sum of all the entries of $W+U$ equals 0. Then the following inequality holds.*

$$Z^* \leq w[V] + \frac{1}{2}\sum_{i=1}^{k} m_i \lambda_i(W+U). \tag{1}$$

Let $\vec{1}_n$ denote the n-dimensional all-ones vector and let $L \in \mathbb{R}^{n\times n}$ stand for the weighted Laplacian matrix of the graph G: $L = \mathrm{Diag}(W\vec{1}_n) - W$, where for some given vector $z \in \mathbb{R}^n$, $\mathrm{Diag}(z)$ stands for the diagonal matrix with order n having vector z for diagonal. Taking for U the diagonal matrix with either $U_{ii} = \frac{-2}{n} w[V]$, for all $i \in [n]$, or $U_{ii} = -\sum_{j:ij\in E} w_{ij}$, for all $i \in [n]$, we deduce the next corollary which provides two upper bounds on Z^* (none of them being dominated by the other for all instances, see also Section 5).

Corollary 1.2.

$$Z^* \leq \frac{1}{2}\sum_{i=1}^{k} m_i \lambda_i. \tag{2}$$

$$Z^* \leq w[V] + \frac{1}{2}\sum_{i=1}^{k} m_i \lambda_i(-L). \tag{3}$$

The problem $\mathcal{P}_m^k$ can be formulated as the following binary quadratic program (see, e.g. [23]), where the columns of the matrix variable X represent the incidence vectors of the node subsets defining the partition.

$$(P)\left\{\begin{array}{lll} Z^* = & \max & \frac{1}{2}\mathrm{Trace}(X^T W X) \\ & s.t. & X\vec{1}_k = \vec{1}_n, \\ & & X^T\vec{1}_n = m, \\ & & X \in \{0,1\}^{n\times k}. \end{array}\right.$$

Barnes et al. [3] present a heuristic to solve $\mathcal{P}_m^k$ that is based on (P). Their method iteratively determines a set of nodes to interchange in order to improve some current partition. This consists in solving a transportation problem that is derived from (P) by replacing the original quadratic objective by a linear approximation whose expression depends on the current solution.

Rendl and Wolkowicz [23] reformulate the problem (P) as an equivalent mathematical program having a linear objective function to be optimized over $(n-1)\times(k-1)$ orthogonal matrices satisfying some linear contraints. From this reformulation, they are able to derive upper bounds on Z^* involving the $k-1$ largest eigenvalues of the matrix $\widehat{W} = V^T W V$, where V is an $n\times(n-1)$ matrix satisfying $V^T V = I_{n-1}$ and $V^T \vec{1}_n = 0$. They also consider perturbations of the main diagonal of W to improve bounds (see [23] for details).

In the statement of $\mathcal{P}_n^k$, removing the restrictions on the sizes of the node subsets defining the partition and taking for the edge weights $w'_{ij} = -w_{ij}$, for all $ij \in E$, we obtain a problem equivalent to the *maximum k-cut problem* which consists in partitioning the node set into k subsets (unrestricted w.r.t. their sizes) so as to maximize the sum of the weights of the edges having their end-points in different parts. For the case when $k=2$, i.e., the so-called *maximum cut problem*, a new class of bounds involving the whole spectral information from the weighted adjacency matrix was introduced in [4, 5]. This approach was generalized in [2] to the maximum k-cut problem, leading to some improvements over other spectral bounds from the literature. In this paper, we are able to extend this work and provide a new class of spectral bounds for $\mathcal{P}_m^k$.

The outline is as follows. In Section 2, we introduce a new class of bounds for $\mathcal{P}_n^k$ involving the whole spectrum of the weighted adjacency matrix. Then, in Section 3, we consider more closely the computation of some geometrical parameters involved in the expression of the new bounds, establishing a connection with a maximum cut problem. In Section 4, we present diagonal perturbations which may lead to some improvements of the bounds. Finally, in Section 5, we report some computational results, before we conclude in Section 6.

2. Spectral bounds

With no loss of generality, we assume the graph G is complete (setting zero weights on non existing edges). In what follows, the inner scalar product is denoted by $\langle \cdot, \cdot \rangle$, and the Euclidean norm by $\| \cdot \|$.

Let $r \in \mathbb{R} \setminus \{1\}$ and $(i, j) \in [k] \times [n]$. Let $\mathcal{B}_i$ stand for the set of vectors in $\{r, 1\}^n$ having exactly m_i entries equal to the value r. Let $d_{i,j}$ denote the distance between $\mathcal{B}_i$ and the linear subspace $\mathrm{lin}(\nu_1, \nu_2, \ldots, \nu_j)$ that is generated by the first j eigenvectors of W:

$$d_{i,j} = \min \left\{ \|z - y\| : z \in \mathcal{B}_i, y \in \mathrm{lin}\left(\nu_1, \nu_2, \ldots, \nu_j\right) \right\}. \tag{4}$$

Theorem 2.1. *Let $r \in \mathbb{R} \setminus \{1\}$. Then, the following inequality holds:*

$$Z^* \leq \frac{1}{2(r-1)^2} \left[\lambda_1 n \left(k + r^2 - 1\right) - 2w[V](2r + k - 2) + \sum_{l \in [n-1]} (\lambda_{l+1} - \lambda_l) \left(\sum_{i \in [k]} d_{i,l}^2 \right) \right] \tag{5}$$

Proof. Let $(V_1, V_2, \ldots, V_k)$ denote a partition of V corresponding to an optimal solution of $\mathcal{P}_m^k$.

For all $i \in [k]$, let the vector $y^i \in \{r, 1\}^n$ be defined as follows: $y_l^i = r$ if $l \in V_i$ and 1 otherwise. We have:

$$\begin{aligned} \langle y^i, W y^i \rangle \quad = \quad & 2r^2 w[V_i] + 2 \textstyle\sum_{j \in [k] \setminus \{i\}} w[V_j] + 2r \textstyle\sum_{j \in [k] \setminus \{i\}} w[V_i, V_j] + \\ & 2 \textstyle\sum_{\substack{(j,l) \in ([k] \setminus \{i\})^2 : \\ j<l}} w[V_j, V_l]. \end{aligned} \tag{6}$$

Let us now compute the sum of each term occurring in the right-hand-side of (6) over all $i \in [k]$.

$$\begin{aligned} &\textstyle\sum_{i \in [k]} 2r^2 w[V_i] && = 2r^2 Z^*, \\ &\textstyle\sum_{i \in [k]} 2 \sum_{j \in [k] \setminus \{i\}} w[V_j] && = 2\,(k-1)\, Z^*, \\ &\textstyle\sum_{i \in [k]} 2r \sum_{j \in [k] \setminus \{i\}} w[V_i, V_j] && = 4r(w[V] - Z^*), \\ &\textstyle\sum_{i \in [k]} 2 \sum_{\substack{(j,l) \in ([k] \setminus \{i\})^2 : \\ j<l}} w[V_j, V_l] && = 2(k-2)(w[V] - Z^*). \end{aligned}$$

Thus, we deduce

$$\sum_{i \in [k]} \langle y^i, W y^i \rangle = 2Z^*(r^2 - 2r + 1) + 2w[V](2r + k - 2). \tag{7}$$

We now derive an upper bound on $\langle y^i, Wy^i\rangle$ making use of the spectrum of W. Before this, we mention some preliminary properties. Note that since W is symmetric we may assume $(\nu_1, \nu_2, \dots, \nu_n)$ forms an orthonormal basis, and consider the expression of y^i in this basis: $y^i = \sum_{l\in[n]} \alpha_l \nu_l$ with $\alpha \in \mathbb{R}^n$. Then, we have $\|y^i\|^2 = \sum_{l\in[n]} \alpha_l^2 = n + m_i(r^2-1)$, which gives

$$\begin{aligned}
\langle y^i, Wy^i\rangle &= \textstyle\sum_{l\in[n]} \lambda_l \alpha_l^2 \\
&= \lambda_1 \left(n + m_i(r^2-1) - \textstyle\sum_{l=2}^n \alpha_l^2\right) + \textstyle\sum_{l=2}^n \lambda_l \alpha_l^2 \\
&= \lambda_1 \left(n + m_i(r^2-1)\right) + \textstyle\sum_{l=2}^n (\lambda_l - \lambda_1)\alpha_l^2.
\end{aligned}$$

From the definition of the distance defined above we deduce $d_{i,j}^2 \le \sum_{l=j+1}^n \alpha_l^2, \forall j \in [n-1]$. Iteratively making use of the inequality $\alpha_j^2 \ge d_{i,j-1}^2 - \sum_{l=j+1}^n \alpha_l^2$ for $j = 2, \dots, n$, we obtain

$$\langle y^i, Wy^i\rangle \ \le \lambda_1 \left(n + m_i(r^2-1)\right) + \sum_{l\in[n-1]} (\lambda_{l+1} - \lambda_l) d_{i,l}^2.$$

Summing these inequalities for all $i \in [k]$, we get

$$\sum_{i\in[k]} \langle y^i, Wy^i\rangle \le \lambda_1 n \left(k + r^2 - 1\right) + \sum_{l\in[n-1]} (\lambda_{l+1} - \lambda_l) \left(\sum_{i\in[k]} d_{i,l}^2\right). \tag{8}$$

Finally, combining (7) and (8), the result follows. $\square$

Remark Enforcing the value '1' among the two possible values for the components of the vectors used in the definition of the distances (4) is done just to slightly simplify the presentation. We are basically interested in the distance between $\mathrm{lin}(\nu_1, \nu_2, \dots, \nu_j)$ and a set of vectors whose components are restricted to take any of two distinct values and must satisfy some cardinality constraints on the number of occurrences of each value. If we denote by d_{j,r_1,r_2} the distance between $\mathrm{lin}(\nu_1, \nu_2, \dots, \nu_j)$ and the set of vectors $\{r_1, r_2\}^n$ with $(r_1, r_2) \in (\mathbb{R} \setminus \{0\}) \times \mathbb{R}$, then $d_{j,r_1,r_2} = |r_1| d_{j,\frac{r_2}{r_1}}$, for all $j \in [n]$, and the results we get by using such vectors are equivalent to the ones presented.

Note that all the terms occurring in the last sum of the inequality (5) are nonpositive, so that removing from the right-hand side some or all of the terms involved in this sum, the expression obtained still provides an upper bound on Z^*, generally weaker than (5), but easier to compute (see Section 3).

The bound (5) on Z^* also raises the problem which consists in determining the best value of the parameter r (i.e., the one minimizing the right-hand side of (5)). Presently, we do not have a general answer for the latter. However, if we consider the following upper bound, obtained from (5) by removing the last term (i.e., the sum over $l \in [n-1]$),

$$Z^* \leq \frac{1}{2(r-1)^2} \left[\lambda_1 n \left(k + r^2 - 1\right) - 2w[V](2r + k - 2)\right], \tag{9}$$

then, one can easily show that the best value for r is $r = 1 - k$. From the computational results reported in Section 5, this seems to be a fairly robust choice.

3. On computing distances

Generally, computing the distances $(d_{i,j})$ is NP-hard. In fact, even fixing $r = -1$, $k = 2$ and taking $m_1 = m_2 = \frac{n}{2}$, the decision problem associated with the problem that consists in determining the single distance $d_{1,n-1}(= d_{2,n-1})$ (for arbitrary vectors $\nu_1, \nu_2, \ldots, \nu_n$ given as input) is NP-complete. This can be shown similarly to the proof of Proposition 4.4 in [4], by reduction from the partition problem with an added cardinality constraint (see [17, p. 223]).

In constrast with this, we show in this section that the distances $(d_{i,1})_{i\in[k]}$ are easy to compute, providing simple closed form expressions for the case of uniform partitions in regular graphs and unit weights. Then, (in Section 3.2), we establish connections between the distance problem (4) and a cardinality constrained maximum cut problem. This will be used in experiments to be described later.

3.1. A particular case: distances of the form $d_{i,1}$

In this section we present a case when distances can be computed efficiently: the distances of the form $d_{i,1}$.

Proposition 3.1. *Let $i \in [k]$ and define I_1 (I_2) as the set of indices in $[n]$ corresponding to the m_i smallest (resp. largest) entries of the vector ν_1. For $j = 1, 2$, let $w^j \in \mathcal{B}_i$ be defined as follows: $w_l^j = r$ if $l \in I_j$, and 1 otherwise. Then $d_{i,1}^2 = n + m_i(r^2 - 1) - \max_{j=1,2} \left\langle w^j, \nu_1 \right\rangle^2$.*

Proof. Let z denote any vector in $\mathcal{B}_i$ and d stand for the distance between z and $\mathrm{lin}(\nu_1)$. We have

$$\|z\|^2 = m_i r^2 + n - m_i = \langle \nu_1, z \rangle^2 + d^2,$$

where the second equation follows from the expression of z in an orthonormal basis of eigenvectors. Since d is minimized when $|\langle \nu_1, z \rangle|$ is maximum, the result follows. $\square$

If G is a connected d-regular graph with unit weights, then the largest eigenvalue of W is d with multiplicity 1 and it is associated with the eigenvector corresponding to the all-ones vector. From Proposition 3.1, we can derive the following closed form expression of the squared distances $(d_{i,1}^2)_{i \in [k]}$ for that case.

Corollary 3.2. *If G is a connected d-regular graph with unit weights, then, $d_{i,1}^2 = m_i \left(1 - \frac{m_i}{n}\right)(r-1)^2$, for all $i \in [k]$.*

The next proposition considers the more particular case of uniform partitions in complete graphs and establishes connections with other bounds from the literature.

Proposition 3.3. *For the case of complete graphs with unit edge weights and uniform partitions (i.e., $m_1 = m_2 = \ldots = m_k = \frac{n}{k}$, assuming n is a multiple of k), the spectral upper bound (5) has value $\frac{n(n-k)}{2k}$, for any $r \neq 1$. And this coincides with the upper bounds (2) and (3) introduced before.*

Proof. From Corollary 3.2, we deduce $d_{i,1}^2 = \frac{n}{k}\left(1 - \frac{1}{k}\right)(r-1)^2$, for all $i \in [k]$. Using this and substituting in formula (5), leads to the first result. We now prove that the spectral bound (5) coincides with the upper bounds given by (2) and (3) for the given particular case. For any b-regular graph we have $\lambda_i(-L) = \lambda_i - b$, where $\lambda_i(-L)$ denotes the i-th largest eigenvalue of the opposite of the Laplacian matrix. Also, since G is complete with unit weights, we have $\lambda_1 = n-1$ and $\lambda_i = -1$, for all $i \in \{2, 3, \dots, n\}$. Using in addition $m_i = \frac{n}{k}$, for all $i \in [k]$ in formulae (2) and (3), the assertion follows. $\square$

3.2. Reducing the computation of distances to maximum cut problems

The squared distance $d_{i,j}^2$ can be expressed as follows: $d_{i,j}^2 = n + m_i(r^2 - 1) - Z_{D0}^*$, where Z_{D0}^* stands for the optimal objective value of

$$(D0)\begin{cases} Z_{D0}^* = & \max \quad z^T V V^T z \\ & s.t. \quad \sum_{l=1}^n z_l = n + m_i(r-1), \\ & \qquad\quad z \in \{1, r\}^n, \end{cases}$$

where $V \in \mathbb{R}^{n \times j}$ stands for the matrix whose k-th column corresponds to the k-th eigenvector ν_k of W. Let us now convert $(D0)$ into an equivalent quadratic optimization problem with variables taking values in $\{-1, 1\}$. To do so, consider the following change of variables:

$$y = \frac{2}{1-r}\left(z - \left(\frac{1}{2} + \frac{r}{2}\right)\vec{1}_n\right) \iff z = \frac{1-r}{2}y + \left(\frac{1+r}{2}\right)\vec{1}_n.$$

Then, problem $(D1)$ becomes

$$(D1)\begin{cases} \max & a^2 y^T M y + 2a y^T M b + b^T M b \\ s.t. & \sum_{l=1}^n y_l = n - 2m_i, \\ & y \in \{-1, 1\}^n, \end{cases}$$

with $M = VV^T$, $a = \frac{1-r}{2}$, $b = \left(\frac{1+r}{2}\right)\vec{1}_n$. Problem $(D1)$ is equivalent to the following problem

$$(D2)\begin{cases} \max & -u^T Q u \\ s.t. & u_0 = 1, \\ & \sum_{l=1}^n u_l = n - 2m_i, \\ & u = (u_0, u_1, \dots, u_n)^T \in \{-1, 1\}^{n+1}, \end{cases}$$

where Q is the symmetric matrix of order $n+1$ having rows and columns indexed by $0, 1, \ldots, n$, and defined by

$$\begin{cases} Q_{00} = 0, & \\ Q_{0l} = Q_{l0} = -ab^T M e_l, & l \in [n], \\ Q_{lm} = -a^2 M_{lm}, & (l,m) \in [n]^2, \end{cases}$$

where e_l stands for the l-th unit vector in dimension n. Problem $(D2)$ can be seen to be equivalent to a maximum cut problem. Consider a complete graph $G = (V, E)$ of order $n+1$, with node set $V = \{0, 1, \ldots, n\}$ and edge weights $w_{lm} = Q_{lm}$, $lm \in E$. Given a node subset $S \subseteq V$, let $\delta(S)$ denote the cut defined by S, i.e., the set of all edges in G having exactly one endpoint in S. Then, the problem which consists in finding a maximum weight cut $\delta(S)$ in G such that $0 \in S$ and $|S \setminus \{0\}| = n - m_i$ can be formulated as follows

$$(D2) \begin{cases} \max & \frac{1}{4}\left(Q_{tot} - u^T Q u\right) \\ s.t. & u_0 = 1, \\ & \sum_{l=1}^{n} u_l = n - 2m_i, \\ & u = (u_0, u_1, \ldots, u_n)^T \in \{-1, 1\}^{n+1}, \end{cases}$$

with $Q_{tot} = \sum_{l=0}^{n} \sum_{m=0}^{n} Q_{lm}$. This is clearly equivalent to $(D2)$.

This connection with the maximum cut problem provides us with a possible approach in order to compute the distances (in particular for dealing with graphs for which $n > 20$, otherwise all the distances may be computed in a few seconds by brute enumeration of integer vectors). Indeed, a branch-and-bound algorithm making use of the following semidefinite program (a relaxation of $(D2)$ strengthened with a cardinality constraint) may be designed.

$$\begin{cases} \max & \frac{1}{4}\left(Q_{tot} - \mathrm{Trace}(QX)\right) \\ s.t. & X_{ll} = 1, l \in \{0, 1, \ldots, n\} \\ & \sum_{l=1}^{n} X_{0l} = n - 2m_i, \\ & \sum_{1 \le l < m \le n} X_{lm} = 2\left(\frac{n}{2} - m_i\right)^2 - \frac{n}{2}, \\ & X \succeq 0, \\ & X \in \mathbb{R}^{(n+1)\times(n+1)}, \end{cases}$$

where the constraint $X \succeq 0$ means that the matrix X is symmetric and positive semidefinite. The formulation can be strengthened with the triangle inequalities and solved by a bundle algorithm, following the approach by Rendl et al. [22]. An important point to be stressed w.r.t. computation times, is that, differently from the experiments reported in [22], the weights are no longer integral in our case. So we cannot conclude with optimality at some node of the branch-and-cut tree as soon as the gap between the upper bound stemming from the relaxation, and some known lower bound is strictly less than one. But we shall rather make use of some precision parameter given as input.

4. Diagonal perturbations

Observe that the upper bound (5) still holds if we modify the diagonal entries of W in such a way that their sum equals 0. Also, the "truncated bound"

$$Z^* \leq \frac{1}{2(r-1)^2}\left[\lambda_1 n\left(k + r^2 - 1\right) - 2w[V](2r + k - 2)\right],$$

which is derived from (5), may suggest to proceed to a modification of the diagonal entries of W so as to minimize the maximum eigenvalue of the resulting matrix. Consider then the following problem.

$$(P0)\begin{cases} \min & -\sum_{l=1}^{n} z_{ll} + \lambda n \\ s.t. & z_{lm} = w_{lm}, \qquad\qquad l \neq m, \\ & \lambda I - Z \succeq 0, \\ & Z \in \mathbb{R}^{n\times n}, \lambda \in \mathbb{R}. \end{cases}$$

In formulation $(P0)$, the matrix Z corresponds to W with possibly nonzero diagonal entries. Observe that in any optimal solution (Z^*, λ^*) of $(P0)$, λ^* always coincides with the maximum eigenvalue of Z^* and $(P0)$ always admits an optimal solution (Z^*, λ^*) such that $\sum_{l=1}^{n} z^*_{ll} = 0$. (Indeed, if $(\widehat{Z}, \widehat{\lambda})$ is an optimal solution of $(P0)$ with $\sigma = \sum_{l=1}^{n} \widehat{z}_{ll} \neq 0$, then $(Z', \lambda') = (\widehat{Z} - \frac{\sigma}{n} I, \widehat{\lambda} - \frac{\sigma}{n})$ is another optimal solution and the trace of Z' equals 0.) So by solving $(P0)$ we can obtain diagonal entries for the matrix W summing up to zero and such that the maximum eigenvalue of the resulting matrix is minimized. Setting $X = \lambda I - Z$, an optimal solution of problem $(P0)$ can be obtained by solving

$$(P1) \begin{cases} \min & \mathrm{Trace}(X) \\ s.t. & x_{lm} = -w_{lm}, \quad l \neq m, \\ & X \succeq 0, \\ & X \in \mathbb{R}^{n \times n}. \end{cases}$$

More precisely, if X^* is an optimal solution of $(P1)$, then an optimal solution of $(P0)$ is given by $(\frac{\sigma^*}{n} I - X^*, \frac{\sigma^*}{n})$, where $\sigma^* = \sum_{l=1}^{n} x^*_{ll}$. In the next section we will illustrate the impact of such diagonal perturbations on the quality of the resulting bounds.

5. Computational experiments

In this section, after we describe the practical setting, we report computational results. First, we study the impact of the parameterizations on the new bounds. We then consider in more detail the computation of bounds for an instance (uniform bisection) taken from the literature. Lastly, we report results on non-uniform partitions.

5.1. Practical setting

All the computational experiments were performed on a laptop using a processor Intel Core i7-2640M CPU @ 2.80GHz x 4, 7.7 Gio RAM. Our implementation is in C language. The SDPs were solved by CSDP [7]. The graphs used in our experiments are as follows, where d stands for a real value in $[0, 1]$.

- $A1, A2$: they correspond to the graphs given in Tables 2 and 3 p.425 from [14], respectively.
- C_n: the cycle with n nodes.
- K_n: the complete graph with n nodes.

- $P_i(n, d)$: a planar graph of order n, with density parameter $d \in [0, 1]$ randomly generated using *rudy* [24] (so that the number of edges is about $3(n-2)d$. Recall that the maximum number of edges of a planar graph with order n greater than 2 is $3(n-2)$) . The index i is an integer to identify a particular graph instance of this type.

- $R_i(n, p)$: a random graph with order n and density parameter d (so that the number of edges is about $\frac{n(n-1)}{2}d$) generated using *rudy* [24]. The index i stands for an identifier for a particular graph instance of this type.

The notation "(W)" close to the name of an instance means that it is edge-weighted, the weights being uniformly and randomly generated in $[-100, 100]$, otherwise all the edge weights have value one. (We indicate in Appendix A the input data to generate the instances different from $A1$ and $A2$ with *rudy* [24].) The upper bounds (2) and (3) are denoted by ubA and ubL, respectively. The upper bounds (5) is denoted by ubS if the original matrix is used and by $ubSD$ if a diagonal perturbation is performed (as described in Section 4).

5.2. The incidence of the parameter r and diagonal perturbations

We report in Table 1 the results obtained on randomly generated instances for the case of uniform 4-partitions. Results on the same instances but for the case of uniform bipartition are deferred in Appendix B. Considering values in the set $\{-k + 0.1q \colon q \in \{0, 1, \ldots, 20\}\}$ for r, the one leading to the best bound, denoted by r_{best} in what follows, was always equal to $1 - k = -1$ for the case of uniform bipartition (so that we do not report it in Table B.4). This contrasts with the results obtained for uniform 4-partitions where some (moderate) improvements may be obtained for some values different from but close to $1 - k$. It is also worth noting that diagonal perturbations may not lead to an improvement of the bound obtained without such perturbation. In fact, in our experiments, it appears that using diagonal perturbations tends to provide better results than the case of no perturbation on instances with random edge weights. A still open question the results suggest is whether there could be some way to determine a priori (i.e., just from the weighted adjacency matrix and possibly its spectrum) whether using diagonal perturbations is the better choice.

Table 1: Upper bounds on Z^* for $k = 4, m_1 = m_2 = m_3 = m_4 = 5$

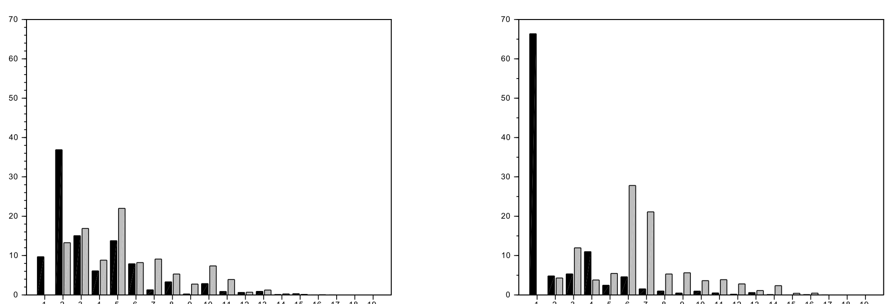
Instance	$\|V\|$	$\|E\|$	ubA	ubL	ubS ($r = 1 - k$)	ubS (r_{best})	$ubSD$ ($r = 1 - k$)	$ubSD$ (r_{best})
$A1$	20	55	35.35	40.35	34.08	34.08 (-3.0)	38.74	38.74 (-3.0)
$A2$	20	51	32.84	40.74	32.64	32.47 (-2.9)	37.05	37.05 (-3.0)
C_{20}	20	20	18.55	18.55	16.06	16.06 (-3.1)	16.06	16.06 (-3.0)
C_{20} (W)	20	20	1179.99	1860.82	932.94	930.79 (-3.1)	920.43	920.43 (-3.0)
K_{20}	20	190	40	40	40	40 (-3.0)	40	40 (-3.0)
K_{20} (W)	20	190	3809.81	4544.49	2479.79	2473.85 (-2.8)	2389.11	2365.44 (-2.5)
$P_1(20, .7)$	20	37	28.68	31.58	28.94	27.64 (-2.5)	30.33	30.33 (-3.0)
$P_2(20, .7)$	20	37	30.81	32.32	30.53	29.53 (-2.5)	30.82	30.82 (-3.0)
$P_3(20, .7)$ (W)	20	37	1622.79	2146.92	1520.15	1442.44 (-2.0)	1055.98	1048.58 (-3.3)
$P_4(20, .7)$ (W)	20	37	1695.25	2040.50	1130.09	1129.40 (-3.1)	957.31	951.54 (-2.5)
$P_5(20, .9)$	20	48	36.05	38.78	35.86	35.48 (-2.7)	37.80	37.80 (-3.0)
$P_6(20, .9)$	20	48	35.08	38.58	34.35	32.79 (-2.4)	36.06	36.06 (-3.0)
$P_7(20, .9)$ (W)	20	48	1714.06	2092.61	1096.65	1096.57 (-2.9)	899.48	888.52 (-2.6)
$P_8(20, .9)$ (W)	20	48	1839.05	2200.52	1521.58	1500.01 (-2.4)	1263.41	1263.41 (-3.0)
$R_1(20, .25)$	20	48	31.03	39.86	29.58	29.19 (-3.3)	34.47	34.47 (-3.0)
$R_2(20, .25)$	20	48	30.47	39.96	30.13	30.13 (-3.0)	34.11	34.11 (-3.7)
$R_3(20, .25)$ (W)	20	48	1771.31	2451.06	1463.36	1439.70 (-2.2)	1214.02	1210.14 (-3.3)
$R_4(20, .25)$ (W)	20	48	1948.73	2727.36	1331.97	1318.41 (-2.6)	1233.05	1230.85 (-2.5)
$R_5(20, .5)$	20	95	45.59	55.27	43.05	42.97 (-3.1)	50.60	50.60 (-3.0)
$R_6(20, .5)$	20	95	42.86	57.29	41.00	40.91 (-3.1)	51.52	51.52 (-3.0)
$R_7(20, .5)$ (W)	20	95	2360.04	3473.43	1914.83	1873.89 (-4.0)	1692.45	1690.44 (-2.6)
$R_8(20, .5)$ (W)	20	95	2635.78	3627.60	2022.28	2022.28 (-3.0)	1810.00	1806.71 (-3.2)
$R_9(20, .8)$	20	152	53.37	63.31	50.56	50.45 (-3.1)	57.47	57.47 (-3.0)
$R_{10}(20, .8)$	20	152	53.03	67.32	51.70	51.50 (-3.1)	62.09	62.09 (-3.0)
$R_{11}(20, .8)$ (W)	20	152	3133.83	4114.32	2543.88	2483.59 (-4.0)	2270.22	2263.82 (-2.8)
$R_{12}(20, .8)$ (W)	20	152	3293.73	4560.00	2258.13	2253.70 (-3.2)	2229.71	2228.03 (-2.9)

Another point that we investigated is: What is the impact of diagonal perturbations on the contribution of each term of the last sum in (5)? This is of interest with respect to "truncated" versions of the bound (5), but also in view of the fact that computing all the distances involved in the bounds is difficult in general. Knowing a priori the most important terms in the last sum of (5), some potentially interesting approximations of this bound can be obtained by computing a limited number of terms. In Figure 1, we represent the percentage of the quantity

$$\frac{(\lambda_{q+1} - \lambda_q)\left(\sum_{i\in[k]} d_{i,q}^2\right)}{\sum_{l\in[n-1]}(\lambda_{l+1} - \lambda_l)\left(\sum_{i\in[k]} d_{i,l}^2\right)},$$

for each $q \in [n-1]$ (horizontal axis) for the instances K_{20} (W) (on the left) and $P(20, .7)$ (W) (on the right) for the case of uniform 4-partitions. It illustrates the relevant terms (and thus also the relevant spectral information) for computing the bounds before (black bars) and after (gray bars) diagonal perturbations. From the experiments that we could carry out so far, when improvements are obtained by perturbing the diagonal, we tend to observe a decrease of the contribution of the first terms. In particular, for the second example illustrated, the contribution of the first term of the sum in the bound (5) becomes zero after diagonal perturbation, whereas it has the largest one before this modification.

Figure 1: The incidence of diagonal perturbations on the terms of the bound (5)



5.3. Some detailed performance on an instance from the literature

We now illustrate the performance of spectral bounds for the graph taken from [11, p. 67]. It is a 40-node and 3-regular graph. In addition to providing us with a comparison between the quality of the bounds obtained using a truncation of (5) and those from [23], the reported results give some indication on the computational effort to determine the distances involved in (5).

Bounds are computed for the case $k = 2$, $m_1 = m_2 = 20$. In Table 2, we mention the value of the bound (5) truncated to the first q terms in the last sum (the ones involving the distances $d_{i,1}, \ldots, d_{i,q}$ for $i = 1, 2$). The diagonal of the weight matrix is perturbed as mentioned in Section 4. It is denoted by $ubSD_q$. We also report the total time needed to compute these bounds (in seconds) and set a time limit of one hour: at some iteration q the computation of the distance $d_{i,q}$ only starts if the current total computation time is less than this limit; but after it has started to compute it, we let it proceed until the distance is computed (hence some times reported exceed this limit). In addition, close to the time we indicate (in brackets) the number of nodes explored in the branch and bound algorithm to compute the distance $d_{i,q}$ (see Section 3). Similarly to [23] we consider three variants of the uniform bisection problem for this graph: one with unit weights ($V1$), and two others $V2, V3$ corresponding the structure of the edge weights $C1$ and $C2$ described in [11, p. 67], respectively. $ubSD_q$ stands for the bound (5) truncated to the first q terms in the last sum and the notation "-" indicates that the time limit was exceeded. The optimal objective values for $V1$, $V2$ and $V3$ are 54, 297 and 322, respectively. The best bounds reported in [23] for these instances are 57.35, 307.10 and 330.42, respectively. So, for all the three cases the spectral bound (5) with modified diagonal entries leads to better results, even when truncated to the first ten terms of the last sum.

Another feature we observe in our experiments is the increase of the computational effort to compute the distances $d_{i,j}$ for an increasing index i. Given some limited amount of time to compute "truncated bounds", this suggests (as done in the experiments described above) to compute the terms of the last sum

Table 2: Upper bounds on Z^* with diagonal perturbation

	$V1$		$V2$		$V3$	
q	$ubSD_q$	Time	$ubSD_q$	Time	$ubSD_q$	Time
0	60.00	0.01	316.00	0.01	341.00	0.01
1	58.61	0.01	310.09	0.01	334.21	0.01
2	58.18	0.09 (3)	308.86	0.02 (1)	332.50	0.04 (1)
3	57.98	2.58 (3)	308.41	0.50 (1)	331.69	3.71 (3)
4	57.73	3.34 (1)	307.34	0.96 (1)	330.81	4.69 (3)
5	57.44	11.80 (7)	307.03	3.75 (7)	330.58	5.04 (1)
6	57.36	22.29 (15)	306.81	5.96 (3)	330.19	6.30 (3)
7	56.92	43.75 (29)	306.15	42.25 (99)	329.35	14.87 (13)
8	56.79	117.49 (219)	305.38	92.34 (155)	328.82	21.33 (19)
9	56.69	311.35 (521)	304.35	177.97 (285)	328.69	36.38 (59)
10	56.60	419.19 (339)	304.16	261.83 (303)	328.46	50.83 (55)
11	56.45	587.42 (599)	303.96	357.52 (373)	328.14	96.73 (159)
12	56.39	902.71 (1211)	303.68	468.59 (457)	327.78	136.14 (171)
13	56.29	1319.13 (1823)	303.59	620.71 (709)	327.48	170.21 (199)
14	56.25	1736.12 (2081)	303.28	868.43 (1237)	327.19	246.18 (375)
15	56.22	2458.07 (3585)	302.97	1263.15 (1849)	327.10	350.17 (551)
16	56.15	3821.50 (6931)	302.76	1614.51 (2023)	326.87	465.41 (699)
17	-	-	302.45	2077.00 (3137)	326.83	536.21 (599)
18	-	-	302.13	2695.19 (4147)	326.72	767.60 (1787)
19	-	-	302.05	3183.48 (3509)	326.38	1203.90 (4075)
20	-	-	301.83	4295.62 (9049)	326.18	1720.67 (4591)
21	-	-	-	-	326.01	2389.04 (6583)
22	-	-	-	-	325.96	3725.91 (13131)

in (5) for an increasing index i.

5.4. Experiments for non-uniform partitions

In order to get some idea of the quality of the bounds (5) for non-uniform partitions, we now consider partitioning the graph $A2$ into two blocks of unequal sizes: m_1 and 20-m_1, for comparison with the spectral bounds from [23] (see Table 8 in this reference). The results are given in Table 3 making use of the following additional notation: $RW1$ stands for the upper bound given by Corollary 4.3 in [23], and $RW2$ for the one given by Theorem 5.1 in [23]. Among the tested values for the parameter r for computing ubS, ($\{r = -2 + 0.1q : q \in \{0, 1, \ldots, 20\}\}$), the best one, leading to the given results, was always equal to $1-k = -1$. Among the different upper bounds $ubSD$ dominates all the others for these evaluations. This raises the question of whether a dominance relation exists, in particular between the bounds $ubSD$ and $RW1$ or $RW2$, at least for some families of instances. This is left for future research. Another feature the results display is an increasing relative gap of the bound $ubSD$ (i.e., the quantity $\frac{ubSD-Z^*}{Z^*}$) when the difference between the sizes of the two blocks increases.

Table 3: Upper bounds on Z^* for non-uniform bipartitions and unit weights

m_1	Z^*	ubA	ubL	$RW1$ [23]	$RW2$ [23]	ubS	$ubSD$
19	50	58.98	50.57	55.71	50.14	50.09	50.01
17	46	56.07	49.72	53.20	48.82	48.09	46.86
15	42	53.17	48.86	49.41	47.80	45.55	44.02
13	40	50.26	48.01	45.87	47.11	43.15	41.93
11	38	47.35	47.16	43.10	46.77	41.26	40.50

6. Conclusion

In this paper we introduced a new class of bounds for graph partitioning. Their expression involves the eigenvalues and eigenvectors of the weighted adjacency matrix. Computational experiments on small instances show they compare well with other bounds from the literature. Computational experiments on larger instances are under work. Future research will focus on the design of a heuristic relying on these bounds in order to compute good-quality solutions and lower bounds.

References

[1] Anjos M.F., Ghaddar B., Hupp L., Liers F., Wiegele A.: Solving k-Way Graph Partitioning Problems to Optimality: The Impact of Semidefinite Relaxations and the Bundle Method. In: Jünger M., Reinelt G. (eds) Facets of Combinatorial Optimization. Springer, Berlin (2013) 355-386.

[2] Anjos, M.F., Neto, J.: A class of spectral bounds for Max k-cut. Research Report. Télécom SudParis, Evry, France. `http://www.optimization-online.org/DB_HTML/2018/12/6401.html` (2018).

[3] Barnes, E.R., Vanelli, A., Walker, J.Q.: A new heuristic for partitioning the nodes of a graph. SIAM J. Discr. Math. 1 (1988) 299-305.

[4] Ben-Ameur, W., Neto, J.: Spectral bounds for the maximum cut problem. Networks 52 (2008) 8-13.

[5] Ben-Ameur, W., Neto, J.: Spectral bounds for unconstrained $(-1, 1)$-quadratic optimization problems. European Journal of Operational Research 207 (2010) 15-24.

[6] Billionnet, A., Elloumi, S., Lambert, A., Wiegele, A.: Using a conic bundle method to accelerate both phases of a quadratic convex reformulation. INFORMS Journal on Computing 29 (2) (2017) 318-331.

[7] Borchers, B.: CSDP, a C Library for Semidefinite Programming. Optimization Methods and Software 11 (1) (1999) 613-623.

[8] Charney, H.R., Plato, D.L.: Efficient partitioning of components. Share/ACM/IEEE Design Automation Workshop, Washington, DC, July 1968.

[9] Chopra, S., Rao, M.R.: The partition problem. Mathematical Programming 59 (1993) 87-115.

[10] Chopra, S., Rao, M.R.: Facets of the k-partition polytope. Discrete Applied Mathematics 61 (1995) 27-48.

[11] Christofides, N., Brooker, P.: The optimal partitioning of graphs. SIAM J. Appl. Math. 30 (1976) 55-69.

[12] Comeau, L.W.: A study of user program optimization in a paging system. ACM Symposium on Operating System Principles, Gatlinburg, TN, October 1967.

[13] de Sousa, V.J.R., Anjos, M.F., Le Digabel, S.: Improving the linear relaxation of maximum k-cut with semidefinite-based constraints. Optimization Online e-print #6567, http://www.optimization-online.org/DB_HTML/2018/04/6567.html (2018).

[14] Donath, W.E., Hoffman, A.J.: Lower bounds for the partitioning of graphs. IBM Journal of Research and Development 17 (1973) 420-425.

[15] Falkner, J., Rendl, F., Wolkowicz, H.: Lower bounds for the partitioning of graphs. Mathematical Programming 66 (1994) 211-239.

[16] Fischer, I., Gruber, G., Rendl, F., Sotirov, R.: Computational experience with a bundle approach for semidefinite cutting plane relaxations of Max-Cut and Equipartition. Mathematical Programming (B) 105 (2006) 451-469.

[17] Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, San Francisco, 1979.

[18] Kernighan, B.W., Lin, S.: An efficient heuristic procedure for partitioning graphs. The Bell System Technical Journal 49 (1970) 291-307.

[19] Lee, J.G., Vogt, W.G., Mickle, M.H.: Optimal decomposition of large-scale-networks. IEEE Trans. System Man Cybernet. 9 (1979) 369-375

[20] Lengauer, T.: Combinatorial algorithms for integrated circuit layout. John Wiley and Sons, Chicester, 1990.

[21] Lisser, A., Rendl, F.: Graph partitioning using Linear and Semidefinite Programming. Mathematical Programming (B) 95 (2002) 91-101.

[22] Rendl, F., Rinaldi, G., Wiegele, A.: Solving Max-Cut to optimality by intersecting semidefinite and polyhedral relaxations. Mathematical Programming (A) 121 (2010) 307-335.

[23] Rendl, F., Wolkowicz, H.: A projection technique for partitioning the nodes of a graph. Annals of Operations Research 58 (1995) 155-179.

[24] Rinaldi, G. Rudy.
https://www-user.tu-chemnitz.de/~helmberg/rudy.tar.gz (1996)

[25] Russo, R.L., Oden, P.H., Wolff, P.K.: A heuristic procedure for the partitioning and mapping of computer logic blocks to modules. IEEE Trans. Comput. 20 (1971) 1455-1462.

Appendix A. Input data to generate the test graphs with "*rudy*"

Instance	Command line
C_{20}	`rudy -circuit 20`
C_{20} (W)	`rudy -circuit 20 -random -100 100 4001`
K_{20}	`rudy -clique 20`
K_{20} (W)	`rudy -clique 20 -random -100 100 2001`
$P_1(20, .7)$	`rudy -planar 20 70 1001`
$P_2(20, .7)$	`rudy -planar 20 70 2001`
$P_3(20, .7)$ (W)	`rudy -planar 20 70 1001 -random -100 100 1001`
$P_4(20, .7)$ (W)	`rudy -planar 20 70 2001 -random -100 100 2001`
$P_5(20, .9)$	`rudy -planar 20 90 3001`
$P_6(20, .9)$	`rudy -planar 20 90 4001`
$P_7(20, .9)$ (W)	`rudy -planar 20 90 3001 -random -100 100 3001`
$P_8(20, .9)$ (W)	`rudy -planar 20 90 4001 -random -100 100 4001`
$R_1(20, .25)$	`rudy -rnd_graph 20 25 1001`
$R_2(20, .25)$	`rudy -rnd_graph 20 25 2001`
$R_3(20, .25)$ (W)	`rudy -rnd_graph 20 25 1001 -random -100 100 1001`
$R_4(20, .25)$ (W)	`rudy -rnd_graph 20 25 2001 -random -100 100 2001`
$R_5(20, .5)$	`rudy -rnd_graph 20 50 1002`
$R_6(20, .5)$	`rudy -rnd_graph 20 50 2002`
$R_7(20, .5)$ (W)	`rudy -rnd_graph 20 50 1002 -random -100 100 1002`
$R_8(20, .5)$ (W)	`rudy -rnd_graph 20 50 2002 -random -100 100 2002`
$R_9(20, .8)$	`rudy -rnd_graph 20 80 1003`
$R_{10}(20, .8)$	`rudy -rnd_graph 20 80 2003`
$R_{11}(20, .8)$ (W)	`rudy -rnd_graph 20 80 1003 -random -100 100 1003`
$R_{12}(20, .8)$ (W)	`rudy -rnd_graph 20 80 2003 -random -100 100 2003`

Appendix B. Computational results on uniform bipartitions

Table B.4: Upper bounds on Z^* for $k = 2$, $m_1 = m_2 = 10$

Instance	$\|V\|$	$\|E\|$	ubA	ubL	ubS ($r = 1 - k$)	$ubSD$ ($r = 1 - k$)
$A1$	20	55	47.13	48.16	43.62	43.50
$A2$	20	51	45.90	46.73	40.04	39.82
C_{20}	20	20	19.51	19.51	18.40	18.40
C_{20} (W)	20	20	1226.39	2270.83	851.53	848.11
K_{20}	20	190	90.00	90.00	90.00	90.00
K_{20} (W)	20	190	4514.77	5565.12	1757.62	1713.28
$P_1(20, .7)$	20	37	37.54	34.30	31.16	30.79
$P_2(20, .7)$	20	37	39.27	34.761	32.00	31.34
$P_3(20, .7)$ (W)	20	37	2046.91	2475.91	1099.56	926.41
$P_4(20, .7)$ (W)	20	37	1988.21	2433.07	691.89	634.12
$P_5(20, .9)$	20	48	48.70	44.63	41.83	41.13
$P_6(20, .9)$	20	48	46.85	42.61	38.60	38.51
$P_7(20, .9)$ (W)	20	48	1957.74	2653.26	570.84	517.17
$P_8(20, .9)$ (W)	20	48	2253.33	2483.39	1329.64	1204.09
$R_1(20, .25)$	20	48	41.06	44.03	36.04	36.79
$R_2(20, .25)$	20	48	42.52	48.00	36.68	36.96
$R_3(20, .25)$ (W)	20	48	2130.71	2795.79	1111.41	1021.17
$R_4(20, .25)$ (W)	20	48	2261.75	3390.18	942.82	889.35
$R_5(20, .5)$	20	95	66.18	73.22	62.28	64.46
$R_6(20, .5)$	20	95	63.34	81.26	59.94	62.54
$R_7(20, .5)$ (W)	20	95	2864.42	4246.53	1382.00	1239.73
$R_8(20, .5)$ (W)	20	95	3278.15	4714.09	1559.22	1462.65
$R_9(20, .8)$	20	152	89.23	97.25	86.15	88.39
$R_{10}(20, .8)$	20	152	89.61	98.99	85.62	87.46
$R_{11}(20, .8)$ (W)	20	152	3663.43	4903.61	1951.99	1810.17
$R_{12}(20, .8)$ (W)	20	152	3737.65	4966.83	1856.24	1736.67